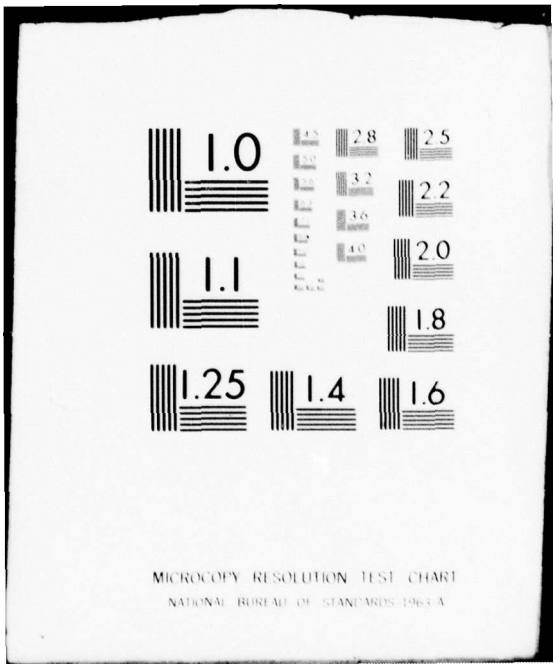
AD-A071 675 PERCEPTION TECHNOLOGY CORP WINCHESTER MASS F/G 17/2
SPEAKER ADAPTATION TEST AND EVALUATION.(U)
MAY 79 H YILMAZ, L FERBER, H KELLETT F30602-77-C-0168
UNCLASSIFIED RADC-TR-79-122 NL

1 OF 1
AD A071675

END
DATE FILMED
8-79
DDC


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963-A

(12) LEVEL II

RADC-TR-79-122
Final Technical Report
May 1979

SPEAKER ADAPTATION TEST AND EVALUATION

Perception Technology Corporation

H. Yilmaz
L. Ferber
H. Kellett

ROME AIR DEVELOPMENT CENTER
Air Force Systems Command
Griffiss Air Force Base, New York 13441

This report has been reviewed by the RADC Information Office (OI) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

RADC-TR-79-122 has been reviewed and is approved for publication.

APPROVED: Richard S. Vonusa

RICHARD S. VONUSA
Project Engineer

APPROVED: H. Davis

HOWARD DAVIS
Technical Director
Intelligence & Reconnaissance Division

FOR THE COMMANDER: John P. Huss

JOHN P. HUSS
Acting Chief, Plans Office

If your address has changed or if you wish to be removed from the RADC mailing list, or if the addressee is no longer employed by your organization, please notify RADC (IRAA), Griffiss AFB NY 13441. This will assist us in maintaining a current mailing list.

Do not return this copy. Retain or destroy.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: RADC-TR-79-122	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle): SPEAKER ADAPTATION TEST AND EVALUATION	5. TYPE OF REPORT & PERIOD COVERED: Final Technical Report
6. PERFORMING ORG. REPORT NUMBER: N/A	
7. AUTHOR(s): H. Yilmaz, L. Ferber, H. Kellett	8. CONTRACT OR GRANT NUMBER(s): F30602-77-C-0168
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Perception Technology Corporation, 95 Cross Street, Winchester MA 01890	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: 31011G, 70550733
11. CONTROLLING OFFICE NAME AND ADDRESS: Rome Air Development Center (IRAA), Griffiss AFB NY 13441	12. REPORT DATE: May 1979
	13. NUMBER OF PAGES: 66
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office): Same (12) 73 p.	15. SECURITY CLASS. (of this report): UNCLASSIFIED
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE: N/A

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

Same

18. SUPPLEMENTARY NOTES

RADC Project Engineer: Richard Vonusa (IRAA)

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Speech Recognition
Pattern Recognition
Acoustic Phonetics

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A speaker adaptable connected word recognizer has been implemented on the basis of a preamble not containing the vocabulary words. The preamble is used to select three closest speakers from the connected speech space of twenty speakers. This acts as a short time adaptation procedure. A long time adaptation procedure is devised where new templates are added from a hard set of connected digits when these digits cause errors. The number of new templates that are needed vary from 2-3 for good speakers to 8-10 for poorer speakers. At the end of this procedure the new speaker often performs equal to 1-3% (Cont'd on reverse)

DD FORM 1 JAN 73 1473

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

below his single speaker connected speech performance.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

EVALUATION

The objective of this program was to develop a real-time and speaker independent algorithm for recognizing a small vocabulary of English words spoken in a natural and unconstrained manner. The algorithm automatically extracts key parameters from a preamble (short phrase different from the vocabulary words) which initiates a speaker transformation which minimizes both inter and intra variations to achieve a speaker independent recognition system.

Overall recognition results for 20 trained and 17 untrained speakers for a vocabulary consisting of the connected digits plus the word "point" spoken in strings of 1, 2, 3 words long, 160 words per speaker were 97.3 and 86.0 percent correct recognition, respectively.

This capability shall impact on future keyword and language recognition programs and have practical applications in data entry, retrieval, and command and control tasks.

Richard S Vonusa

RICHARD S. VONUSA
Project Engineer

INTRODUCTION.

Programs are written to build a speaker- and channel-independent connected speech recognizer for 21 words (digits 0 through 9 and point, plus 10 command words). The speaker independence is realized on the basis of a preamble not containing the vocabulary words. The recognizer is built on the new PDP 11/70 computer. The evaluation and data base were accumulated using new recordings and new template making and optimization procedures. The real-time operation, pitch correction, saturation normalization, and speed normalization efforts necessitated entering into an exploratory mode where an increasing number of directions had to be experimented. The development time was extensive because some of these directions did not work out satisfactorily and others resulted in a compromise in accuracy.

1. To achieve real-time operation an attempt was made, as soon as the new machine became operational, to replace template-matching by spectral correlation with a spectral subtraction procedure (mean-error minimization). The spectral subtraction method is computationally much faster than correlation since it involves no multiplications. However, the end results are less satisfactory.

2. Because of its inherently slow running speed, a Fortran program does not lend itself to real-time operation. With an array processor the system would run in real-time when all programs are converted into machine language. This is very costly in time and money and was not pursued, especially since all the system problems are not satisfactorily solved. At present, the

time window is 2.5 seconds. This accommodates at most 7 connected digits when uttered rather fast. Due to the limited memory handling capability of the computer, it was impractical to process utterances longer than 2.5 seconds. This time window is, however, sufficient to utter 7 digit telephone numbers when spoken fast. At normal rates of speaking 5 to 6 digits are easy to accommodate. Otherwise the program is able to handle an unlimited number of words in a string.

3. Speed normalization (rate of speech per unit time) is implemented by adding to the previous time normalization a time-warp algorithm where selected samples in the utterance correlating with the successive selected samples to yield a score above a preset value of correlation parameter are skipped. From observations made this performs reasonably well. This speed normalization requires in general more samples in the utterances than in the templates. This is accomplished by sampling the utterances in the recognition mode more frequently than in the template making mode. Templates are made using .92 as the sampling parameter, while during recognition the data is sampled with .96 as the sampling parameter. Again the performance is quite satisfactory, although by going to .92 value of the sampling parameter some discrimination loss seems to have occurred.

4. After the time-warp algorithm, the editing rules had to be reexamined and the program updated. A good feature was that it was possible to reduce the extraneous errors (errors of the type 3 → 3,8 where 8 is extraneous) considerably. As a result

the present recognizer makes fewer extraneous errors.

5. Extensive attempts were made to implement pitch correction. Early in time a hardware commercial pitch extraction system was purchased and tried. It extracts the pitch only from long stable regions but it was not possible to extract it from transition regions and near voice-unvoice regions where we needed it most. After some effort it became clear that it would not work satisfactorily in the intended way.

6. Originally, the contract required 10 digits. Later the word "point" was added at the suggestion of the agency for reasons of flexibility in handling material containing digits. The word "point" often confuses with some of the digits. This reduces considerably the accuracy. Without "point" in the data base the accuracies would be considerably higher. (The largest number of errors come from "point" confusing with 1 and 4.)

7. The zero-crossing circuitry, a hardware component, required an additional channel which was not available in the new machine's 16-channel data acquisition system. As a result, the voice-unvoice feature became less reliable. The voice-unvoice feature is necessary in averaging over the skeletons so that selected samples do not get out of step. An automatic averaging process was tried and the results were not completely satisfactory. The speaker independence was then pursued in the direction of using the preamble to identify one to three closest speakers in the data base and using for recognition the templates of these speakers. This procedure is workable and it is much simpler than the expansion method, both

in the creation of the data base and in updating during the long-time adaptation process.

8. To achieve speaker and channel independence the concept called for averaging over large numbers of vowels to obtain the basis functions and averaging over a large number of skeletons (voiced samples with voiced, unvoiced samples with unvoiced). Also, skeletons belonging to templates taken from the hard set should be averaged among themselves according to the position from which they are taken. A large vocabulary with most words multi-syllable presents a very large number of alternatives, with many representatives from each position. To average by hand and by human judgement became impossible. Attempts were made to do so automatically and this did not produce satisfactory results. The conclusion here seems to be that careful averaging over a large number of basis functions is necessary to obtain the average basis functions and averaging over a large number of skeletons is necessary to obtain the average skeletons across speakers. After this is done some sort of category formation is also needed to put similar dialects and accents into separate categories. With the time available to us after the programs became operational, this was not possible to carry out.

9. The short-time adaptive procedure based on preamble KEY SUE FUR SHOP is used to identify the closest 1 - 3 speakers to the unknown person's speech. When the templates of these 1 - 3 speakers are used for recognition a short-term adaptation is achieved. Thereafter a long-term adaptation is applied during

which a few new templates are added from a hard set of digits (such as 411, 383 etc.) or from errors that occur as the system performs. Results are close to the results obtained for single speaker connected mode recognition. In this mode singles usually perform with none or very few errors. The templates to be added are mostly for the connected digits

10. The complete statistics presented are a combination of adaptive recognition (7 speakers) and single speaker recognition results (37 speakers). The errors and their analyses are given separately in the results section. This summarizes the test results on 44 speakers (men + women). It is to be stressed that only one sample of a word was stored from each speaker. Time did not permit each speaker to train on 10 or more utterances of each word. Taking into consideration all the limitations under which work was done, the end results are quite satisfactory and indicate that the speaker-adaptable recognizer of connected words is a real possibility along these lines. With a CRT at his disposal a good speaker can be trained within 20-30 minutes to a consistency of 95-98% accuracy in the connected mode. Some familiarity with speech patterns and how to make templates are, however, necessary since full automation of the procedure has not been possible at the present time. The results should be viewed under the consideration that more than half of these speakers were completely untrained and were unfamiliar with any kind of recognition task.

11. The procedures developed, both for short-time and long-time adaptation, are independent of speaker and channel; that is,

the procedures are applicable to both equally well. The filter system bandwidth is essentially very close to the telephone bandwith. Because there was no array processor or time to convert the programs into machine language the short-time adaptation process uses a few minutes of calculation time. However, to utter the preamble words takes less than 10 seconds. With a fast computation scheme the total would be no more than 10 seconds. The long-time adaptation and updating at present takes 50-60 minutes because templates are made by hand. With automatic template making this would take about 60 seconds, depending on how many of the templates need to be added or updated.

12. The recognition algorithm is independent of the vocabulary chosen or the phrase to be recognized, since any word or phrase can be used or added to the vocabulary. It is also independent of the length or number of syllables in the phrase, since each syllable can be included in a string of syllables which sequentially represent the phrase and recognized by this sequential representation.

13. The present performance and error analysis reflect the major source of errors as: a) Voice-Unvoice became less reliable, hence the large number of errors in "six". b) The word "point" confuses with "one" and "four". c) More than half of the speakers are either novices or have dialects and accents.

14. In the near future it is planned to redo the basis functions on the basis of averages of large numbers of individual basis functions and transform with these averaged functions. It is also

planned to reintroduce the zero-crossing feature and reexamine the time-warp algorithm. In this way, the present programs are likely to be made to work effectively. In the meantime the present adaptive algorithm should be considered satisfactory since it performs reasonably well and since it is simple to implement and operate.

1. RECOGNITION EXPERIMENTS

With the above recognition algorithms, several experiments were performed to test the various speaker-adaptation options. Whereas the initial efforts were directed toward implementing spectral adaptation, the predominant direction was speaker categorization.

1.1 Spectral Adaptation.

Attempts were made to perform spectral speaker adaptation (by the methods discussed in Chapters 2 and 3) using adaptation of the voiced parts of the templates with 4 vowel functions, and subsequently using 6 and 8 basis functions (voiced and unvoiced) to adapt all template samples.

1.1.1. Experiment To Build Up an Adequate Master-Skeleton File with 4-Vowel Adaptation.

a) Goal. To amass a minimal data-base of skeletons made from a limited data-base of recorded utterances, and which are sufficient to perform recognitions of single digits (0-9) when a new speaker enters his/her basis functions.

b) Basis functions used. Basis functions are 5 instances per speaker of I, @, A, and U from the utterance "He Had Hot Food" repeated 5 times. Each instance of a vowel consists of 16 filtrates at the times of two selected samples at the peak of each vowel. Basis functions are stored as templates in an ordinary template file. They are selected using a program that edits out all selected samples not at vowel peaks; this program should be instrumental in automating the vowel-extraction process.

c) Method of Making Skeletons. For each vowel name (I, @, A, U), a 16-vector basis function is constructed as described in Section B.6.1. Each unvoiced selected sample in each word-template is stored directly (without alteration) as the corresponding selected sample of the skeleton of the word-template. Each voiced selected sample is expanded into a skeleton of coefficients which are stored as real numbers where the 16 byte-filtrates used to be in the template file. (Since 1 real number = 4 bytes, 4 real numbers fit exactly into 16 bytes.) The skeletons are now devoid of spectra of the speaker's voice (except for unvoiced regions).

d) Method of Reconstituting Templates. The new speaker's vowel basis functions are obtained as in Sections B.6.1 and B.6.3. Each unvoiced selected sample is stored directly as the corresponding selected sample of the reconstituted word-template. For each voiced selected sample, the four skeleton coefficients are multiplied by the respective basis functions of the new speaker. The sum of these four products (a 16 byte-vector) is the reconstituted voiced selected sample, and is stored where the skeleton coefficients were in the template file. The reconstituted templates are now endowed with the voice spectra of the new speaker (except for unvoiced regions).

e) Methods of the Experiments. Basis function template files were made for 5 speakers (HK, LF, DD, DM, MB), and thirty single-digit utterances in random order were stored on disk for each of these speakers. Templates for the single digits (0-9)

were made for speaker HK (from new utterances not on the disk). These templates were expanded into skeletons, reconstituted with HK's basis functions, and then run in a recognition against HK's 30 single-digit utterances. Templates were made from representative uttered digits missed in the recognition run. (By "representative" is meant that if several instances of "five" are missed, only one of them is made into templates.) These templates were expanded into skeletons, added to the existing skeleton file, and then the whole file was reconstituted with HK's basis functions. This was repeated until speaker HK had no error in the recognition run.

Now a new speaker was introduced. The skeleton file was reconstituted in terms of LF's basis functions and a recognition was run on LF's utterances. Skeletons were made from representative utterances missed in recognition via LF's basis functions. This process was once again iterated until no errors were made. New people were added to the data base until no errors occurred in the next speaker's utterances when run against a reconstituted template file made only from skeletons from previous speakers (and, of course, reconstituted via the new speaker's basis functions). Finally, the skeleton file was recycled over all the data-base speakers to make sure no errors had accumulated due to reconstituted template competition.

f) Results. The sequence of single errors was as follows:

HK - 5 errors initially--removed by alternative 0,6,8

LF - 2 errors initially--removed by alternative 2,3

DD - 3 errors initially--removed by alternative 9,1,0

DM - 3 errors initially--removed by alternative 0, 9, 7.

MB - no errors; cycle complete

Recycle - no errors.

These results were encouraging, but could not be consistently obtained in subsequent trials. Errors in such words as "six" indicated that unvoiced regions also needed spectral adaptation. Therefore, unvoiced basis functions were introduced in the experiments that followed. Due to the unreliability of voice-unvoice mentioned before, a larger number of skeletons from different speakers could not be averaged meaningfully. Speaker categorization was therefore implemented on the basis of one to three closest speakers.

1.2 Speaker-Categorization Adaptation.

Although the tests of the speaker-categorization method of adaptation (described in detail in Section B.7) may properly be regarded as experiments, they are not interim experiments that paralleled and instructed the development of the recognizer. Rather, they are tests of the completed recognition algorithm, performed in an "open loop" without further modification of the algorithm. Therefore, these test results are presented separately in Section 2.

2. RESULTS

The results are presented in two stages. The first stage consists of the tests that were performed while constructing the data base, the second stage is the speaker independent test. The results are presented in tables showing the performance of each speaker. They are also analyzed using confusion matrices and normalized error plots.

2.1 Data Base and Test Material.

The material for the test consisted of strings of digits and command words recorded for fortyfour (44) speakers, thirtysix (36) male and eight (8) female. Each speaker was given a random list of digits to read in a connected manner. The list was made in such a way that all digits had an equal representation but in random combinations. Each speaker read 30 single digits, 20 utterances of double digits (40 digits) and 30 utterances of triples (90 digits) for a total of 160 digits. In the context of this experiment the digits are the English digits "zero" through "nine" and the word "point". In addition, for the purpose of data base generation and speaker adaptation a second set of recordings was recorded for each speaker. This recording consisted of a set of adaptation utterances, a single repetition of the digits and a "hard set" of digit utterances. The hard set consists of a subset of the digits in such combination that they present problems due to coarticulation. Each speaker read the same list (which is presented in Section 3.7).

2.2 Data Base Generation.

The data base was constructed using 29 male and 8 female speakers. One set of digits, spoken in a discrete fashion from each speaker and one set of command words were used. A set of templates, one for each word, was made for a total of 37 sets of templates. This was the preliminary data base and was the starting point for the second stage of data-base updating. The updating was done on the "hard set" by running a recognition test using the preliminary templates. The errors were corrected by the addition of templates until the "hard set" of digits reached satisfactory performance. This procedure was repeated 37 times until the preliminary set of templates for each speaker was updated to the point of acceptable performance on the "hard set" of recordings.

2.3 Performance of Data Base Speakers.

To test the performance of the data-base speakers, a recognition run was performed on the random utterances of each speaker. Since the template file in each case was that speaker's file, this is a test of a single speaker connected word recognition system. The object of this test was to evaluate a practical system for the recognition of the digits, the word "point", and ten other command words when used in a simulated data-entry environment. Even though the system is a single-speaker system, it can be considered practical in a multispeaker environment since only a single repetition of the vocabulary read in a discrete fashion was used for initial training.

The results are summarized in Tables 1 through 4. The wide variation in performance is due to the fact that most speakers (31 out of 37) never talked to a speech recognition system before and tended to slur their words when reading connected strings. Table 1 was arranged in descending order of performance. It is clear that for the top 20 speakers, the performance is substantially higher than for the remaining 17 speakers, 97.3% and 86.0% respectively. Tables 3 and 4 summarize the confusion matrices showing the errors and the contribution of the individual words to total number of errors. Each confusion matrix contains the total number of errors per word, the number of extraneous words and the number of rejections per word. In Tables 2 through 4, a "?" means the word was missed, an "ex" means the word was printed as an extra word and the "P" denotes the word "point". The words that contributed most to the errors were the words "six", "eight", and "point". This is true for the top 20 speakers as well as the bottom 17 speakers, indicating that the voice/unvoice categorization failed for both categories of speakers.

2.4 Recognition of the Control Words.

A set of 10 control words was entered into the data base using the same 37 speakers. The words; enter, retrieve, continue, plus, recall, minus, mistake, backspace, reset & stop, were stored as templates. The templates were based on a single repetition of these words by each of the 37 data base speakers and no additional corrections were used. To test the performance, each

No.	Speaker	Sex	No. of Errors	% Correct
1	RMA	M	0	100.0
2	IFE	M	0	100.0
3	GAM	M	0	100.0
4	LFE	M	2	98.8
5	DCO	M	2	98.8
6	AJO	M	2	98.8
7	BKE	M	3	98.1
8	BBE	M	4	97.5
9	DDE	M	4	97.5
10	JRU	F	4	97.5
11	MBR	M	5	96.9
12	DMC	M	6	96.3
13	ENA	M	6	96.3
14	MKO	M	6	96.3
15	TSI	M	6	96.3
16	JWE	M	7	95.6
17	HKE	M	7	95.6
18	SKE	F	7	95.6
19	OCA	F	7	95.6
20	WSA	M	8	95.0
21	AFE	M	14	91.3
22	FHO	M	15	90.6
23	RDA	M	15	90.6
24	EMC	M	15	90.6
25	BKE	M	17	89.4
26	TST	M	17	89.4
27	KOU	F	18	88.8
28	HYI	M	19	88.1
29	SEV	M	19	88.1
30	RHA	M	19	88.1
31	BMA	M	20	87.5
32	PTE	F	20	87.5
33	NBI	F	26	83.5
34	FKO	M	30	81.3
35	SGR	M	32	80.0
36	KSV	F	40	75.0
37	ECU	F	44	72.5

TABLE 1.

A summary of recognition results for a vocabulary of 11 words, the digits and the word "point". The table is arranged in order of decreasing performance.

RECOGNIZED

SPOKEN	0	1	2	3	4	5	6	7	8	9	P	?
0		1	2	1	1		1	1		1		1
1					5	1				3	2	6
2	1			9			3	2	13			27
3		1	8				6	2	14	1		15
4	6	8		2		1		2				22
5		3			3			1		3	5	14
6	7		5	3	2			4	5			60
7	4		2		5	1	1		3			14
8			1	7			6	1				31
9		21			2	2						3
P		19		3	21	2	1		6	2		3
ex			1		1				19			

Table 2.

Confusion matrix for 37 speakers male and female, showing the errors for connected digits and the word point. A total of 5920 words were used in strings of 1, 2, 3 words per string. Overall accuracy including errors from all sources was 92.1%.

RECOGNIZED

SPOKEN	0	1	2	3	4	5	6	7	8	9	P	?
0												
1					1	1				1		2
2				1			1		2			4
3			2						3			1
4		1										2
5								1			3	1
6				1					5			8
7					1							
8			1	3			3					5
9		2			1	1						
P		10			7					1		
ex			1						9			

Table 3.

Confusion matrix for 20 trained speakers. The connected digit plus the word point were spoken in strings of 1, 2, 3 words long, 160 words per person. Overall accuracy including errors of omission, commission and extraneous words is 97.3%.

RECOGNIZED

SPOKEN	0	1	2	3	4	5	6	7	8	9	P	?
0		1	2	1	1		1	1		1		1
1					4					2	2	4
2	1			8			2	2	11			23
3		1	6				6	2	11	1		14
4	6	7		2		1		2				20
5		3			3					3	2	13
6	7		5	2	2			4				52
7	4		2		4	1	1		3			14
8				4			3	1				26
9		19			1	1						3
P		9		3	14	2	1		6	1		3
ex					1				10			

Table 4.

Confusion matrix for 17 untrained speakers. The connected digits and the word point were spoken in strings of 1, 2, 3 words long, 160 words per person. Overall accuracy including errors of omission, commission and extraneous words is 86.0%.

of the speakers read the words one word at a time in random order. In each case the templates used were that speaker's own templates. A summary of the results is shown in Table 5. The results are based on 20 words per person for 37 speakers for a total of 740 words. There were a total of 15 errors, 13 of which were rejection errors for a score of 98.0% correct recognition.

2.5 Speaker Adaptation Test, Using Categorization.

To evaluate the performance of the system for a set of speakers that were not in the data base, the following procedure was used: Each speaker reads the preamble "KEY SUE FUR SHOP". The system performs a correlation of the preamble with all preambles of the 37 data base speakers. The highest scoring speaker or speakers are chosen as representatives and their templates are used as a reference vocabulary for recognition purposes. The algorithm will select 1, 2 or 3 categories depending on a distance measure among the top 3 candidates in the data base. In the present system each of the 37 speakers represents a category. This is sub-optimal, since by cross-correlation of the data base a set of categories can be found which will reduce the number of categories and make them more representative. The test was performed for seven speakers; six speakers from PTC and one speaker from RADC. The RADC speaker (speaker RV) was tested as part of the final demonstration.

The results are shown in Table 6. After the selection of a category the test speaker reads the "hard set" of digits. When an

RECOGNIZED

SPOKEN	Plus	Enter	Reset	Stop	Continue	Backspace	Mistake	Recall	Retrieve	Minus	?
Plus				1							1
Enter											1
Reset											2
Stop											2
Continue											
Backspace							1				
Mistake											2
Recall											4
Retrieve											1
Minus											

Table 5.

Confusion matrix for the command words showing the errors for a data base of 37 speakers. There were 15 errors out of 740 words for a 98.0% correct recognition.

error occurs, the templates in that category are augmented to eliminate that error. The number of errors in the "hard set" for each speaker is shown in the second column in Table 6. This is also the number of additional templates that were added to that category. The speaker then reads the list of test utterances. The number of errors and % correct recognition (columns 3 and 4) indicate performance after the "hard set" augmentation. The test material was based on data recorded for the seven speakers following the same procedure and the same data structure as the material for the other 37 speakers.

Speaker	No. Errors "Hard Set"	No. Errors Test Set	% Correct Recognition
HK	13	7	95.6
MB	3	7	95.6
AD	2	2	98.8
LF	16	8	95.0
FK	15	5	96.9
RM	7	6	96.3
RV	9	6	96.3

Table 6.

Results for 7 Speakers Using the Adaptation by Category Method.

3. CONCLUSIONS.

The results of the present investigation indicates that speaker adaptation by way of category formation can be made to work with a combination of short-time and long-time adaptation. During the short-time adaptation, a preamble not containing the vocabulary words is made to select a subset of templates from the overall data base. The new speaker usually performs with very few errors on singles. On doubles and triples, errors are higher. During a long-time adaptation, a few of the templates, either from the hard set of connected words or from the positions of errors, are corrected. At the end of this process the speaker performs close to his single speaker performance. With the short-time adaptation an operator can start using the machine and, as time goes on, with the long-time adaptation, can bring his performance to 98.8% - 95% accuracy. The latter process was carried out for 7 male speakers. It is noteworthy that for these 7 speakers the single digits plus "point" performed with 99.05% correct recognition.

REFERENCES

1. Yilmaz, H. (1967), "A Theory of Speech Perception", Bulletin of Mathematical Biophysics 29, 793-825.

2. Yilmaz, H. (1968), "A Theory of Speech Perception II", Bulletin of Mathematical Biophysics 30, 455-479.

3. Yilmaz, H. et.al.(1976), "Automatic Speaker Adaptation", Final Report RADC-TR-76-273 (F30602-75-C-0227.), AD# A032592.

APPENDIX A.

A. CONCEPTUAL BACKGROUND

A.1 Color Analogue in Spectral Adaptation to Speech.

Our algorithm for speaker-independent speech recognition begins by addressing a more specific problem, that of removing the speaker dependence from the sound-energy spectrum of a steady-state vowel. The problem has been discussed from a psychophysical point of view by Yilmaz (1967, 1968), and from a pattern-recognition point of view by Yilmaz et al (1976).

From a standard acoustical argument, a vowel's sound-energy spectrum can be expressed as the product of an energy spectrum $I(\lambda)$ from the vocal apparatus (including larynx and cavity resonators) and a modulating spectrum $R(\lambda)$ from the articulators. We hypothesize that most of the speaker dependence of the vowel resides in the vocal apparatus, and the modulating spectrum conveys the identity of the vowel.

Thus the problem of representing spoken vowels in a speaker-independent way becomes analogous to that of recognizing object spectral reflectances independently of the spectrum of the incident light. The vocal-apparatus sound-energy spectrum $I(\lambda)$ (an excitation function) is analogous to the illuminant energy spectrum, and the modulation $R(\lambda)$ from the articulators is analogous to reflectance. The analogy is particularly clear when we view selective reflection of light from a surface as two-way (entering and leaving) transmittance of the light through the translucent layer constituting the surface. Viewing spectral reflectance as a transmittance

clarifies the physical analogy with the selective-transmittance properties of the articulators in the vocal tract.

As is the case with reflectance spectra, we also hypothesize that a typical articulator modulating spectrum tends to be slowly-varying in wavelength, so it can be approximated by an expansion in terms of four basis functions $r_k(\lambda)$*:

$$R_i(\lambda) = \sum_{k=1}^{4} \alpha_{ik}\, r_k(\lambda) \qquad (2\text{-}1)$$

The same speaker-independent coefficients α_{ik} (called the skeleton of the vowel) characterize the expansion of the vowel-sound filtrates through the 16 filters $q_j(\lambda)$ of the PTC recognizer:

$$F_{ij}(I) = \sum_{k=1}^{4} \alpha_{ik}\, f_{kj}(I) \qquad (2\text{-}2)$$

where $F_{ij}(I) \equiv \int q_j I\, R_i d\lambda$, and $f_{kj}(I) \equiv \int q_j I\, r_k d\lambda$.

The recognition proceeds as follows: Calculate the skeleton of vowel i using the filtrates $F_{ij}(I)$ for this vowel and filtrates $f_{kj}(I)$ for the basis vowels k, all spoken by a control speaker I. Then have a new speaker J say a preamble in which the basis-vowel filtrates are identified and recorded, and construct an estimate of vowel i said by the new speaker. In order for a candidate vowel spoken by J to be recognized as the vowel i, its filtrate 16-vector

---

* The number 4 was arrived at by finding empirically that four expansion functions are sufficient to construct intelligible speech in a vocoder; further expansion functions add only prosodic qualities to the reconstructed signal.

(later to be reduced to 4 independent expansion functions) must match the estimate for vowel i obtained by the above adaptive method. In this way, an arbitrary vowel can be recognized by a new speaker without a prior utterance of that vowel by the new speaker.

In the above procedure, it is necessary to solve Equations (2-2) to find the skeleton α_{ik} of vowel i. One needs only four of these 16 equations to find the skeleton: The system is over-determined. We resolved this ambiguity by computing the α_{ik} that rendered a least-squares best fit of $\sum_k \alpha_{ik} f_{kj}(I)$ to $F_{ij}(I)$. (i.e., we minimized

$$\sum_{j=1}^{16} [F_{ij}(I) - \sum_{k=1}^{4} \alpha_{ik} f_{kj}(I)]^2 \qquad (2\text{-}3)$$

by the standard method.)

The least-squares method involves solving for α_{ik} the equations

$$\sum_{j=1}^{16} F_{ij}(I) f_{j\ell}(I) = \sum_{k=1}^{4} \alpha_{ik} \left(\sum_{j=1}^{16} f_{kj}(I) f_{j\ell}(I) \right) \quad \ell = 1,4$$

which is equivalent to using as audio response functions the basis functions $f_{j\ell}(I)$ of the speaker in question. If these basis functions are spiky in their spectra, however, small errors in their assessment will be magnified by letting them play the role of response spectra in determining the skeleton. The computed skeletons may be more stable with respect to basis-function errors if we use smoothly-varying orthogonal functions as the

response functions of the system. For example, in place of $f_{j\ell}(I)$ in the above equation, we might insert 1, $\sin(\frac{2\pi j}{16})$, $\cos(\frac{2\pi j}{16})$, $\sin(\frac{4\pi j}{16})$. These functions are similar to the response functions in the color theory that motivated this approach. (Orthogonality is not absolutely necessary in the theory, but aids the computational accuracy.)

A.2 Assumptions Particular to Speech.

So far, we have assumed that the speaker-dependent driving function $I(\lambda)$ is constant in time for a single speaker. However, since loudness can change from one word to the next in a speaker's utterance, it is more reasonable to relax this assumption so that the driving function can vary in amplitude with time, but remains constant in relative spectral composition. Thus for two speakers I and J, $I(\lambda,t) = g_I(t)I(\lambda)$ and $J(\lambda,t) = g_J(t)J(\lambda)$. (Note that even this relaxed assumption does not yet take into account pitch variations within a single speaker's utterance.)

Suppose t_{Ik}, t_{Jk} are the times at which vowel k is extracted from the utterances of speakers I and J. Similarly, let a test vowel V be uttered by the two speakers at times t_{IV}, t_{JV}, respectively. Then the filtrates (j=1, 16) measured by the PTC recognizer for vowel V will be

$$\frac{F_{Vj}(I)}{g_I(t_{IV})} = \sum_{k=1}^{4} \frac{\alpha_{Vk}}{g_I(t_{Ik})} f_{kj}(I) \qquad (2\text{-}4)$$

and similarly for speaker J with I replaced by J.

Without access to the energy envelopes $g_I(t)$ and $g_J(t)$, one cannot directly infer the skeleton coefficients α_{Vk} from the filtrates $F_{Vj}(I)$, $F_{Vj}(J)$, $f_{kj}(I)$, $f_{kj}(J)$. One gets only the quantities

$$\alpha_{Vk} \frac{g_I(t_{IV})}{g_I(t_{Ik})} \equiv \beta_{Vk}(I) \qquad (2\text{-}5)$$

and similarly for speaker J.

We cannot solve this problem by artificially normalizing the filtrates in each 16-vector spectrum (as in peak normalization), for the normalization factors will not generally compensate for the loudness changes. (After all, the envelope to be normalized is on the driving function, not on the peak amplitudes of the product spectra corresponding to the basis vowels.) Theoretically, we could resort to finding the envelope ratio between speakers directly by taking the ratio between basis-function spectra corresponding to the same vowel name from the two speakers. However, the PTC filters are fairly broad-band compared to the spectral transitions of characteristic speech sounds at any given time. Thus such a ratio can be done only on paper, not by the machine. Making the PTC filters narrow-band will only degrade their time resolution, so this problem is significant and not an accidental property of the present recognizer. Thus we have to adopt an approach that allows the 16 PTC filters to sum the spectrum before further processing is done on the resulting filtrates. The basis-function approach can be made to do this as follows:

Define a fiducial spectrum for each speaker corresponding to a

known vowel that is not one of the basis functions. Denote it by subscript 0, so that

$$\frac{F_{0j}(I)}{g_I(t_{I0})} = \sum_{k=1}^{4} \frac{\alpha_{Vk}}{g_I(t_{Ik})} f_{kj}(I) \qquad (2\text{-}6)$$

and similarly for speaker J.

Then, as before, one can infer by least-squares best fit the quantities

$$\beta_{0k}(I) = \frac{\alpha_{0k}\, g_I(t_{I0})}{g_I(t_{Ik})} \qquad (2\text{-}7)$$

and similarly for speaker J.

Before we proceed further, we note that each of the spectra we are approximating will be peak-normalized (scaled so the maximum component is 255) before being compared to similar spectra in an unknown utterance. Therefore, we can without loss of generality assume any factor we want to scale the spectra $F_{Vj}(I)$, $F_{Vj}(J)$. In particular, we can assume

$$g_I(t_{I0})/g_I(t_{IV}) = 1 = g_J(t_{J0})/g_J(t_{JV})$$

so that

$$\frac{\beta_{Vk}(I)}{\beta_{0k}(I)} = \frac{\beta_{Vk}(J)}{\beta_{0k}(J)} = \frac{\alpha_{Vk}}{\alpha_{0k}} \qquad (2\text{-}8)$$

is speaker-independent.

Suppose now that we use the utterances of speaker I to compute $\beta_{0k}(I)$, $\beta_{Vk}(I)$. Then, after finding $f_{kj}(J)$, $\beta_{0k}(J)$ for a new speaker J, we can predict (up to a scaling factor) the spectrum

for V spoken by J from the relation:

$$F_{Vj}(J) = \sum_{k=1}^{4} \frac{\beta_{Vk}(I)}{\beta_{0k}(I)} \beta_{0k}(J) \; f_{kj}(J) \qquad (2\text{-}9)$$

This is the theory of speaker-independent vowel reconstruction so far as we see it. The $\beta_{0k}(I)$ is introduced for compensating intensity variations during basis-function extraction. In the process of optimizing the recognition performance of the word-recognizer, it has been expedient to include unvoiced sounds as well as vowels in the expansion; we settled on the sounds I, E, A, U, S, $ for later experiments, but earlier experiments just used I, @, A, U to reconstitute voiced sounds. In all experiments, basis-functions were obtained from spectra averaged over from 3 to 5 utterances and this should be extended to at least 10 utterances for purposes of greater smoothness of the basis-functions.

We are now developing a version of the recognition algorithm that incorporates the basis-function normalization via $\beta_{0k}(I)$. We believe that incorporating our understanding of basis-function normalization will significantly improve the recognition results on adapted word templates. In the future, we will also average the skeletons β_{Vk}/β_{0k} from different utterances to remove any residual speaker-dependences. Up to now, we have had difficulty performing such averaging because of a voice/unvoice detector which could not be relied upon to determine voice/unvoice boundaries from which selected speech samples could be sequentially averaged.

APPENDIX B.

B. METHODS OF IMPLEMENTATION

In this section, we shall describe the operation of the recognition system developed and tested by Perception Technology Corporation.

B.1 Initial Signal Processing.

The system prepares all incoming signals for further processing in the manner described below:

B.1.1. Fixed Interval Samples.

The system is designed to process an utterance of duration 2.5 seconds at each entry. The length of this time span is imposed only by the capacity of the computer. During the allowed 2.5 seconds, approximately five single syllable connected utterances can be entered. The signal is passed through a bank of 16 weighted analog filters covering the range of audio spectrum. A description of characteristics of these filters is given in Section B.9. The system is triggered manually or automatically before each entry. Upon triggering the system starts taking readings of the 16 filters every 10 milliseconds, giving a maximum of 225 fixed interval samples. (Automatic triggering in a long utterance proceeds in 2.5 second pieces consecutively staggered backwards by 0.5 seconds to encompass vocabulary words eclipsing the end of each time window.)

Following each reading of the 16 filters, calculations are performed on the data during the 10 millisecond interval before another set of readings is taken. Two quantities are obtained as follows:

a) A_i, sum of filter amplitudes, for sample number i.

$$A_i = \sum_{j=1}^{16} f_{ij}$$

where f_{ij} is the amplitude of the output from the j th filter when sample i was taken. Preliminary energy normalization scales all the energies so that the maximum over a 2.5 second window is set equal to 5000 (a maximum convenient for integer storage of the energy values). Referring to Figure 1A, i denotes the sequential numbers on the left-most column and A_i is represented by the position of the symbol *.

b) R_i, square root of the sum of squares of f_{ij}, is given by

$$R_i = [\sum_{j=1}^{16} |f_{ij}|^2]^{1/2}$$

R_i is needed for subsequent correlation calculations. The numbers f_{ij}, A_i, R_i are all stored in memory.

B.1.2. Noise Level and Detection of Beginning and End of Signal.

Silence portions are monitored for the purpose of determining the noise level. The noise level is taken to be the time average of noise energy in the channel during the absence of speech. The threshold level, V_t, for a signal is defined as 1.5 times the average energy of the first ten noise samples:

$$V_t = 1.5 \, [\sum_{j=1}^{16} \sum_{j=1}^{16} f_{ij}]/10$$

During microphone input, the program notes the first superthreshold sample in the 2.5-second window, and continues processing until a sample with a sub-threshold energy occurs. If this sample is less than 6 samples after the first superthreshold sample, the program looks for another first superthreshold sample. Otherwise, the program marks the first subsequent instance of a superthreshold sample followed by .4 seconds of subthreshold energy. At this point, the samples extending from .2 seconds prior to the first superthreshold sample to .4 seconds after the final superthreshold sample are included in the utterance to be processed subsequently.

As an example, in Figure 1A, the beginning sample No. is 57 and the ending sample No. is 214. The whole utterance to be processed lies between these two samples and can contain many syllables.

B.1.3. Voice and Unvoiced Classification.

Contained among the fixed interval samples are those representing vowels, consonants and gaps, each of which is to be classified as either voiced or unvoiced. A sample i is classified as unvoiced if the following two conditions are simultaneously satisfied:

a) The energy A_i of the system is less than 32% of the maximum energy in the 2.5 second utterance.

b) The ratio of the summed energies of the 4 lowest-frequency filters to the summed energies of the 4 highest-frequency filters is less than or equal to .6. This algorithm is the software replacement of the zero-crossing criterion used in the previous contract.

B.1.4. Peak Normalization.

In connected speech, the voiced maxima have varying amplitudes even within a 2.5 second window, reflecting changes in loudness as the speaker talks. In order to have a meaningful index of the rate of change of energy, we peak-normalized each voiced region longer than 6 time-samples. The normalization is a quadratic function of the energy $G(A_i)$ such that the voiced maximum A_{max} maps (always up) into 5000, and also

$G(0) = 0$ (Zero energy is preserved.)

$G'(0) = 1$ (Attack and decay rates are independent of the utterance's peak loudness, when evaluated near the beginning and end of the peak, at energy minima of nearly zero energy. This reflects a natural property of speech sounds.)

Subject to these constraints, the normalization mapping is

$$G(A_i) = A_i + \frac{A_i^2}{A_{max}} \left(\frac{5000}{A_{max}} - 1\right)$$

B.2 Selection of Normalized Samples.

The final samples are selected from the fixed-interval samples as follows:

B.2.1. Time Normalization.

The basic idea of time normalization is to sample the speech so as to render it more or less independent of the rate of speaking. This is usually done in a crude way by taking the samples only after a significant amount of spectral change occurs. This, however, is not sufficient because the

intensity , the rate-of-change of intensity, voicing etc. are part of the recognition criteria. The time-normalization was therefore improved by including the following factors:

a) The amount of change in spectral shape:

The amount of spectral change between two samples i and i' is measured by their correlation C_{ii}

$$C_{ii'} = \frac{\sum_{j=1}^{16} [f_{ij} \cdot f_{i'j}]}{[\sum_{j=1}^{16} |f_{ij}|^2]^{1/2} [\sum_{j=1}^{16} f_{i'j}|^2]^{1/2}}$$

b) The change in energy:

The change in energy between two samples i and i' is measured by the absolute value

$$|A_{i'} - A_i|$$

c) The nature of the signal i.e., voiced or unvoiced.

d) The level of signal energy.

e) The duration of the signal.

B.2.2. The Use of Parameters.

Parameters are used to associate the factors mentioned above with a weight and they can be readily employed to evaluate the relative influence of each factor. The voicing parameter is used to label the voiced samples as 1 and unvoiced samples as -1. In the present sampling procedure we have utilized spectral correlations and intensity changes to measure changes starting from

the first sample that exceeds the threshold. All these variables are monitored sequentially and the combined correlations are calculated through the use of these parameters. When the combined correlations drop below a preset criterion, say, $\xi = 0.96$, a sample is selected. Starting from the selected sample this process is repeated until the whole utterance is exhausted.

The combined spectral correlations and intensity changes between two samples i and i' are calculated as follows:

$$\text{Combined correlations} = C_{ii'} - D_c \cdot |A_{i'} - A_i|$$

where D_c is the parameter weighing the relative importance of intensity change with respect to spectral change. In general, in an utterance of three connected numerals, the total number of such samples varies between 30 and 40. The numbers are nearly independent of time but vary with habit and dialect. In Figure 1A, the column labelled N gives the normalized samples selected by the system. The dashed lines are proportional to $C_{ii'}$, for $i = 60$, the sample with the peak energy.

B.3 Method of Performing Matches Between Templates and an Unknown Utterance.

Matches to a template are found by "walking" the template through the unknown utterance and finding the best correlation of selected samples in the template with selected samples in the utterance. The best correlation for a given tentative position of the first selected sample in the template is found by a method of

time-warping, as follows (see Fig. 2): The tentative position of the template-match in the utterance is marked by the first selected sample of the template. The second template sample chooses the best-correlating of the next two utterance samples, and latches onto it. The third template sample chooses between the two utterance samples following the utterance sample chosen by the second template sample. The matching proceeds in this way until the last sample of the template is matched, and then a cumulative correlation is computed with the chosen samples in the utterance. This is the template score.* Since the piece of the utterance that matches the template always has more selected samples (possibly as many as twice the number in the template) it is necessary to choose fewer selected samples in the template than in the corresponding utterance: The sampling speed must be reduced for template-making.

At each selected sample in the utterance, the names and scores of the three best-matching templates starting at that selected sample are stored, together with the corresponding lengths of the matching utterance in selected samples. Template scores that are below a template-dependent threshold are reset to zero.

B.4 Editing Rules for Evaluating Word Matches.

The editing program stitches together the above template matches to produce word identifications. To do this, it first scans the template-matching data in temporal order until it finds template matches that are appropriately ordered and spaced (within a tolerance of one selected sample) that constitute a likely instance of a vocabulary

* Actually, all scores are correlations multiplied by 200.

word. A word score is computed as the mean of the (1 to 6) constituent-template scores, and word scores below a word-dependent threshold are reset to zero. After this, the program looks for later instances of any vocabulary word, but skips the part of the utterance already matched by the sequence of templates in the completed identification.

Because this stitching algorithm favors matches at the beginning of the utterance that eclipse later, higher matches, a correction called creep is introduced. This delays finalizing a word identification until the program scans downstream one-third the length of the utterance segment corresponding to the tentatively-identified word. If it finds a higher score, the new word preempts the old one; this process is iterated until the end of the utterance, if necessary. (See Fig. 1B for a sample recognition plot.)

For the vocabulary of the present contract, we often found a spurious "eight" riding in the wake of identified words such as "three". To eliminate such errors (facilitated by the shortness of the typical "eight" template) all putative "eights" had to satisfy one of the following conditions:

a) The word score is greater than 180.

b) The energy increases sometime during the putative "eight".

c) The energy of at least one selected sample exceeds 2/5 of that of the nearest voiced-peak maximum.

B.5 Speed-Up of The Template-Matching Routine.

The most time-consuming part of the recognition algorithm is the correlation of the selected samples of the template with the

unknown utterance. This is so because the operation must be done so many times, and also because the correlation measure--involving many multiplications--is intrinsically time-consuming to compute. We tried two methods to reduce this time, and thus bring the recognition closer to real-time.

Our first attempt was to replace the correlation with a simpler measure of the difference between compared spectra. We tried the mean error (the sum of the absolute differences of the filtrates of the compared spectra) because it has no multiplications at all. Although time was saved by this method, our recognition results deteriorated to an unacceptable level. Undoubtedly this was because much of the rest or our recognition algorithm was predicated on the use of the correlation. Therefore, we returned to the correlation in the interests of saving system development time.

Our second attempt was more successful: We sought to reduce the number of times the correlation was executed by rejecting template-matches after a few low-scoring selected samples were correlated. The template correlation was abandoned at sample 1 if the correlation at sample 1 was less than ZTHR; it was abandoned at sample 2 if the correlation for the first two samples was less than ZTHR-ZINC; it was abandoned at sample n if the correlation for the first n samples was less than ZTHR-nZINC. The final value of ZTHR-nZINC was the template threshold. By establishing this initially rigorous but subsequently relaxing criterion for continuing correlation, we were able to save 30-40% of the recognition time.

B.6 Speaker Adaptation by Spectral Reconstitution.

Our initial attempts at speaker-adaptation during the period of this contract were guided by the theoretical principles of Section 2: We sought to remove from word templates the spectra of the data-base speaker's voice and insert the voice spectra of the new speaker. The method we employed had three basic parts:

B.6.1. Making Skeletons From a Data-Base Speaker's Word Templates and Basis-Function Templates.

For each vowel name (I, Ö, A, U), and for each unvoiced-sound name (S, $)* (extracted from the preamble "Key Sue Fur Shop"), a 16-vector basis function was constructed by averaging the selected samples in each basis-function template with that name, and then averaging the three instances of the sound. Averaging over more instances would have produced smoother basis functions similar to color theory's response functions. For each selected sample in each word template, a least-squares best fit was found with a linear combination of the basis functions. The six coefficients in this expansion are stored as integers where the 16 byte-filtrates used to be in the template file. (Since 1 integer = 2 bytes, 6 integers fit into 12 of the 16 bytes allocated for the filtrates of a selected sample of speech.) The expansion coefficients form a skeleton that is devoid of the spectra of the speaker's voice.

Our first experiment with spectral adaptation used only the vowels I, @, A, U, (taken from the preamble "He Had Hot Food") to adapt the voiced part of the template, and simply carried the un-

* K and F were not reliably extracted.

voiced part through without adaptation (see Section 2.1.1.). However, we found that the unvoiced speech also needs adaptation; since our voice/unvoice detector is unreliable, we expanded every template sample in terms of all 6 spectra I, Ö, A, U, S, $ (taken from the preamble "Key Sue Fur Shop"), instead of trying to segregate and expand separately the voiced and unvoiced parts of the template. (Ultimately, we expect greater reliability from the expansion when we can reliably partition the expansion domain into subspaces with distinct voiced and unvoiced basis functions.)

As shown in Fig.3, program SKLTRN (a utility that is separate from the main COMMON task) operates on a master template file (consisting of all speakers' templates) to produce the master skeleton file. In the process, each speaker's basis-function templates are brought to bear on his/her word templates.

B.6.2. Reconstituting Templates From Data-Base Word Skeletons And a New Speaker's Basis Functions.

The new speaker's basis functions were obtained as above from his/her basis-function template file. For each selected sample in each skeleton, the six skeleton coefficients were multiplied by the respective basis spectra of the new speaker, and the results were added together to produce a 16-byte vector. This vector is stored where the skeleton coefficients were in the template file (and extending into the vacant areas where the old templates used to be before skeleton creation). The reconstructed templates are now endowed with the voice spectra of the new speaker. The program that does this is OUTTMP (see Fig. 3).

B.6.3. Extracting Basis Phonemes From a Training Preamble.

For each data-base speaker and also for each new speaker, a template file was made containing the basis phonemes I, U, Ö, A, S, $ from three utterances of "Key Sue Fur Shop". For the vowel phonemes (I, U, Ö, A), each template consisted of two selected samples nearest the voiced energy peak of the relevant word. For the unvoiced sounds (S, $), each template consisted of the two unvoiced selected samples nearest the voiced onsets of "Sue" and "Shop", respectively. Occasionally the $ phoneme would register at least one voiced sample because of high energy and the unreliable low-versus-high frequency discrimination of the voice/unvoice detector. In such cases, the $ generally creates a distinct voiced energy peak; therefore, the $ was extracted from this peak when it occurred.*

B.7 Speaker Adaptation by Speaker Categorization and Template-File Augmentation.

The results of recognition experiments performed on the spectrally adapted templates were not entirely satisfactory, because the basic theory was still developing, the expansion seemed sometimes to degrade the template spectra, and the voice/unvoice detector was unreliable. Also, the projected time required to process the 50 data-base speakers via spectral adaptation exceeded the time to complete the contract reasonably. Therefore, we adopted a method of speaker adaptation by categorization. This method involves immediate adaptation to a new speaker by switching in a template file for the most similar 1 - 3 speakers in the data base; the subsequent long

* In some experiments (Section 2.1) 4 or 8 basis functions were used instead of the 6 described here.

term adaptation is implemented by adding new templates from the new speaker to correct recognition errors. The details of the method are as follows:

B.7.1. Short-Term Adaptation by Speaker Categorization.

Templates for the 21-word vocabulary are stored for each of 44 data-base speakers in separate template files. Also, templates from an utterance of "Key Sue Fur Shop" are stored from each data-base speaker. This constitutes the data entered prior to the new speaker's introduction to the machine.

A new speaker begins by saying "Key Sue Fur Shop" a single time into the microphone, and the computer goes through the following steps in about two minutes:

a) The "Key Sue Fur Shop" is automatically segmented into 4 parts and matched against the "Key Sue Fur Shop" of each data-base speaker.

b) The three best correlation scores are recorded, and the templates from the three best-matching speakers are entered into a single template file (via a special task COMBIN).

c) This template file is installed for subsequent word recognition from the new speaker.

B.7.2. Long-Term Adaptation by Adding Templates From Recognition Errors to the New Template File.

After the initial short-term adaptation, the new speaker utters the following triplets of digits deemed difficult to recognize by the machine: 118, 111, 311, 318, 418, 411, 711, 718, 911, 918, 831, 838, 819, 841, 848, 849, 859, 088, 188, 288, 388, 488, 788, 888, 988. Similar utterances are spoken for the command words. These

utterances are stored automatically with sequential utterance-file names, and then recalled for automatic recognition. An operator notes the recognition errors, and makes templates from them. The recognition process takes about five minutes, and the template-making (done manually via the interactive graphics terminal) takes no more than a minute for each word error.

If this process is repeated at various intervals during a long session of the new speaker with the machine, it can correct for slow variations in utterance manner, and thus is a form of continuing, long-term speaker adaptation.

B.8 Software Overview.

Our programming efforts were devoted to transferring recognition programs from the PDP 8 to the PDP 11/70 running under the RSX-11M operating system. Midway through the contract period, we changed from Version 2 to Version 3 of the RSX-11M, which is the version we are presently using. We have developed programs to enable the system to store utterances automatically and to interact conveniently with human operators by means of a light pen and a CRT graphics terminal. Progress to date is described below:

B.8.1. Operating System.

By replacing Version 2 of the RSX-11M with Version 3, we obtained greater versatility during program development, and also attained an efficient interface with human operators using a light pen and a VT-11 graphics terminal. We also increased processing speed without sacrificing versatility by discontinuing the use of the PDP 11/10 as an adjunct. We were thus able to dispense with DECNET,

which was a cumbersome--though reliable--mode of coupling the 11/10 with the 11/70.

There are three principal tasks in the recognition program; all three exist in memory together and are handled in parallel. Within each task, there is also an intricate memory management scheme between the subroutines and the system common blocks. This required a significant amount of time to implement, but has rewarded us with an increased speed and versatility in the processing.

Subroutines of each task are automatically switched in and out of memory by system event flags and system common blocks. Human intervention occurs only through the graphics terminal, which displays options on a CRT and allows manipulation via typewriter commands and via a light pen.

B.8.2. Task Structure.

There are three principal tasks in the present recognizer:

COMMON — Written in Fortran IV Plus for optimum execution times and for the added disk-handling features, this task handles most disk files related to the project. These include template files, word files, utterance files, scratch files, and various utility files used by the task. COMMON also performs all time-dependent algorithms used for recognition. It receives all its instructions from the GRAFIC task, via system event flags and parameters left in the system common blocks.

GRAFIC — This task performs all functions needed to interface the operator with the project. The functions include menu selection, graphic display of results, and control of the COMMON task.

PRNT — This task prints files created by the COMMON task. Its operation is much like that of the print spooler of the RSX-11M system. The PRNT task will also send reproduced speech to a speaker, when the operator wishes to hear stored taped data.

The internal structures of the COMMON and GRAFIC tasks are summarized in the program-structure diagrams (Figs. 4, 5).

B.8.3. Program Structure of The COMMON Task.

The COMMON task consists of a number of program modules, connected as indicated in Fig. 4. The function of each of the modules is summarized below.

COMMON — Controls the program flow for the entire speech program. It starts the graphic handler (the GRAFIC task), and waits until GRAFIC returns with a menu selection (e.g., to train the recognizer or to use it to perform recognition. Then it decides which program section to call.

INITIT — Performs all the once-only initialization for the COMMON main program.

DIRECT — Manipulates word and template files, according to the user's specification of which words and

templates are to be active (i.e., candidates for utterance matches).

EXITIT — Performs an orderly exit from COMMON.

DUMP — Dumps common blocks on the lineprinter, if the user so desires. DUMP is called through HELP, which will soon become a more general-purpose help module.

TRAIN — Controls program flow for the section of COMMON that makes templates from input utterances.

INPUT/DINPUT — Reads input data from microphone and disk, respectively. (Speech read from the disk has already been converted to digital form.) These input routines also perform some processing of the data. Whereas the raw data are the energy filtrates from 16 filters evaluated at up to 250 ten-millisecond intervals, the INPUT routines pass these data to the rest of the program in four forms: The BAUDIO array contains all the filtrates, but for each sampling time the filtrates are normalized so the maximum of the 16 filtrates evaluated at that time is 255; the ILVSUM array contains the sum of the filtrates evaluated at each sampling time, normalized so the maximum over the 2.5 second window is 5000; the ILOSUM and IHISUM arrays are respectively the sums of the four lowest frequency and four highest frequency filtrates, evaluated at each sampling time.

VOICE1 Evaluates the state of voicing at each sampling time, depending on the relative values of ILOSUM and IHISUM. Thereafter, the speech is divided into voiced, unvoiced and gap.

NORM2 Normalizes the ILVSUM in each voiced region to 5000, and performs a smoothing function such that artificial discontinuities are not thereby introduced into the speech data. NORM2 also eliminates short voicing-bursts such as are characteristic of noise.

SAMPLE Evaluates the speech data in order to choose selected samples at times when the speech is changing most rapidly. SAMPLE removes some of the contingency on length of utterance, by performing an effective time-normalization on the speech signal.

SAVTPL Saves templates created in TRAIN, by storing them on the disk.

PLOT Plots speech data (including ILVSUM and selected samples) on the lineprinter, for a permanent record that affords easy visual access.

RECOG Controls program flow for the section of COMMON that performs recognition of words in an unknown utterance, based on matching templates in the data base to parts of the utterance. Recognition is performed after the unknown utterance is passed through the input routine, and samples are selected (as with the templates).

RECSUB — Stores the unknown utterances, and then packs the selected samples into the initial buffers of BAUDIO and ILVSUM (ILOSUM and IHISUM are not used after the VOICE1 routine). The latter buffers of these arrays are used to store templates, which are cycled through as candidates for recognition in RECOGA.

RECOGA — Takes candidate templates from activated spot words, and moves them through the unknown-utterance selected samples, looking for a match. The best matches are found by a method of time-warping, as discussed in Section B.3.

RECOGB — Performs word recognitions from the templates recognized in RECOGA. The three best-recognized templates starting at each selected sample of the unknown speech are passed from RECOGA to RECOGB. A word is recognized starting at a given selected sample if, within a tolerance, the sequence of templates in that word can be recognized in the right order without interfering with one another. A word score is then computed, which is the average of the correlations of the constituent templates with the unknown speech. The best recognition, if its score exceeds a threshold, is accepted as a word recognition.

AUTOST — Performs automatic scanning though long utterances in 2.5 second pieces, in order to achieve automatic recognition.

ADAPT — Matches the new speaker's preamble ("Key Sue Fur Shop") against that of all data-base speakers, and combines into a single file the templates of the 3 best-matching speakers. This file is then switched in for subsequent recognition of the new speaker's utterances. Sub-module ADAPTA does the categorization, and ADAPTB switches in the new file.

The functions of the modules of the GRAFIC task, which is slaved to the COMMON task, reflect the functions of the corresponding modules in COMMON.

In addition, there is a task (PLYTSK) that interpolates spectra between the selected samples of an utterance and uses the resulting numbers to drive sound sources so the utterances can be heard via an audio speaker. Another task, COMBIN, combines the template files selected in speaker categorization (See Section B.7.1.).

B.8.4. Auxiliary Utilities.

We found it most convenient to separate some of the subroutines from the main COMMON program, in the interests of simplifying the graphic menus where speed and contiguity were not essential (particularly in the training phase of the program). The principal utility tasks are listed below:

STORIT — Stores utterances in files coded automatically by speaker. This program requests a speaker number via the DEC-writer and then uses a bell to prompt

the speaker to make sequential 2.5 second utterances into the microphone. The stored utterances are used to test recognition performance, and also include the preamble "Key Sue Fur Shop".

SKLTRN — Expands the 16-vector spectra at each selected sample of each spot word template in terms of a least-squares approximation by a linear combination of basis functions characteristic of the speaker of the template. The coefficients are the speaker-independent skeleton of the template.

OUTTMP — Reconstitutes speaker-independent template-skeletons by multiplying the skeleton coefficients by a new speaker's basis functions and adding the products to produce the appropriate linear combination of basis functions. The reconstituted templates are now endowed with the voice spectra of the new speaker.

EDTTMP — Allows quick editing of template files, including renaming, re-ordering, deletion, and directory listing. Wild card options facilitate the editing operation.

B.9 Hardware Overview.

The word recognition system consists of a Digital Equipment Corporation (DEC) PDP 11/70 computer with 128K of memory, other DEC-supplied peripheral devices, custom-made audio filters, and recording devices. The configuration is shown in Fig. 6.

B.9.1. DEC-Supplied Peripheral Devices.

The standard peripheral devices are as follows:

1	RP04	Disk drive (44 megabytes)
2	RK05	Disk drive (1.2 megabytes)
1	TU16	Magnetic tape drive (9 track, 1600 BPI maximum)
1	LP-11	Lineprinter
1	VT-11	Graphics display system (GT-42)
1	AR-11	Analog real time system
1	DR-11C	General device interface
3	DL-11	Asynchronous serial line interface

Initially, the VT-11 was configured in its own PDP 11/10 computer. At that time, this required DECNET (DEC's network software) to communicate between the 11/10 and the main computer (11/70). DECNET proved to be very cumbersome to use, so the VT-11 was installed in the PDP 11/70, which now handles the graphics directly. The only disadvantage of this arrangement is that the VT-11 slows the central processor considerably when it is displaying graphics. Because of this, the VT-11 is turned off when program running speed is important. Recently, DEC has released a more efficient network software package, which we plan to use to distribute the graphic task once again to the PDP 11/10.

B.9.2. Audio Filter Assembly (custom-made).

The audio signal is first amplified and passed through a pre-emphasis network and a band-pass filter. The pre-emphasis network is an active RC network providing 6 dB/octave pre-emphasis between 700 and 4000 Hz, an emphasis near 300 Hz, and de-emphasis just below 700 Hz. The signal is band-limited by a bandpass filter with 24 dB/octave slopes and 6 dB-cutoff frequencies of 250 Hz and 5300 Hz.

After pre-emphasis, the signal is sent through 16 data channels, each having an active filter, an active full-wave rectifier

(with a 60 dB dynamic range), and a low-pass smoothing filter. The active filters have different characteristics, but the rectifiers and low-pass filters are all the same. The latter are 2 pole RC filters with a 3-dB cutoff frequency of 15 Hz. The active filters for the 16 channels are two-stage multiple-feedback filters with the following characteristics:

Filter No.	Center Frequency (Hz)	Q
1	260	5.00
2	317	5.00
3	387	5.00
4	472	5.00
5	576	5.00
6	694	6.35
7	812	6.35
8	950	6.35
9	1111	6.35
10	1300	6.35
11	1600	5.00
12	1952	5.00
13	2381	5.00
14	2905	5.00
15	3545	5.00
16	4325	5.00

Inputs to the filter bank can come from the microphone or can be generated from a D/A converter connected to the DR-11C interface. The output from the D/A converter is fed through a low-pass filter to minimize quantization noise.

B.9.3. Pitch Extractor and Voice/Unvoice Detector.

A commercial McMorrow pitch extractor was purchased during the contract period, but it was useful only for steady-state sounds atypical of real speech. Therefore, it was not used in the present system. Also, a hardware voice/unvoice detector (including a zero-crossing ciruit) which had functioned

effectively with our previous PDP 8-based recognizer, was not incorporated into the present recognizer because it would have required a system overhaul for which we could not afford time.

FIGURES

```
** PLOT ** TRAINING PLOT  FOR ROME

FVOWEL=  12.00  FRATIO=   0.60  FACLV=0.00000500  CHANGE=
 0.960 IRECTH=   130

MAGNIFY START =    0 MAGNIFY END =    0

DATE : 21-JAN-79   TIME : 20:48:07

START SAMPLE =    57   FINAL SAMPLE =   214


  57 *    1 -* ----------
  58 *    0 -* ----------
  59      0 -* ----------
  60      0 -* ----------
  61      0 -* ----------
  62      0 -* ----------
  63      0 -* ----------
  64      0 -* ----------
  65      0 -* ----------
  66      0 -* ----------
  67      0 -* ----------
  68      0 -* ----------
  69      0 -* ----------
  70      0 -* ----------
  71      0 -* ----------
  72      0 -* ----------
  73      0 -* ----------
  74      0 -* ----------
  75      0 -* ----------
  76 *    2 -* ----------
  77      0 -* ----------
  78 :    0 -* ----------
  79 :    3 -* ----------
  80 :    0 -* ----------
  81 :    4 ---* ----------
  82 :    5 -----* ----------
  83 :    6 --------* ----------
  84 :    7 ----------* ----------
  85 :    0 ----------* ----------
  86 :    8 ----------* ----------
  87 :    0 ----------* ----------
  88 :    9 ----------* ----------
  89 :    0 ----------* ----------
  90 :    0 ----------* ----------
  91 :   10 ----------* ----------
  92 :    0 ----------* ----------
  93 :   11 --------* ----------
  94 :    0 -----* ----------
  95 :   12 ---* ----------
  96 :    0 --* ----------
  97 :   13 --* ----------
```

Fig. 1A. SAMPLE TRAINING PLOT, "ONE-TWO-THREE" (First Section)

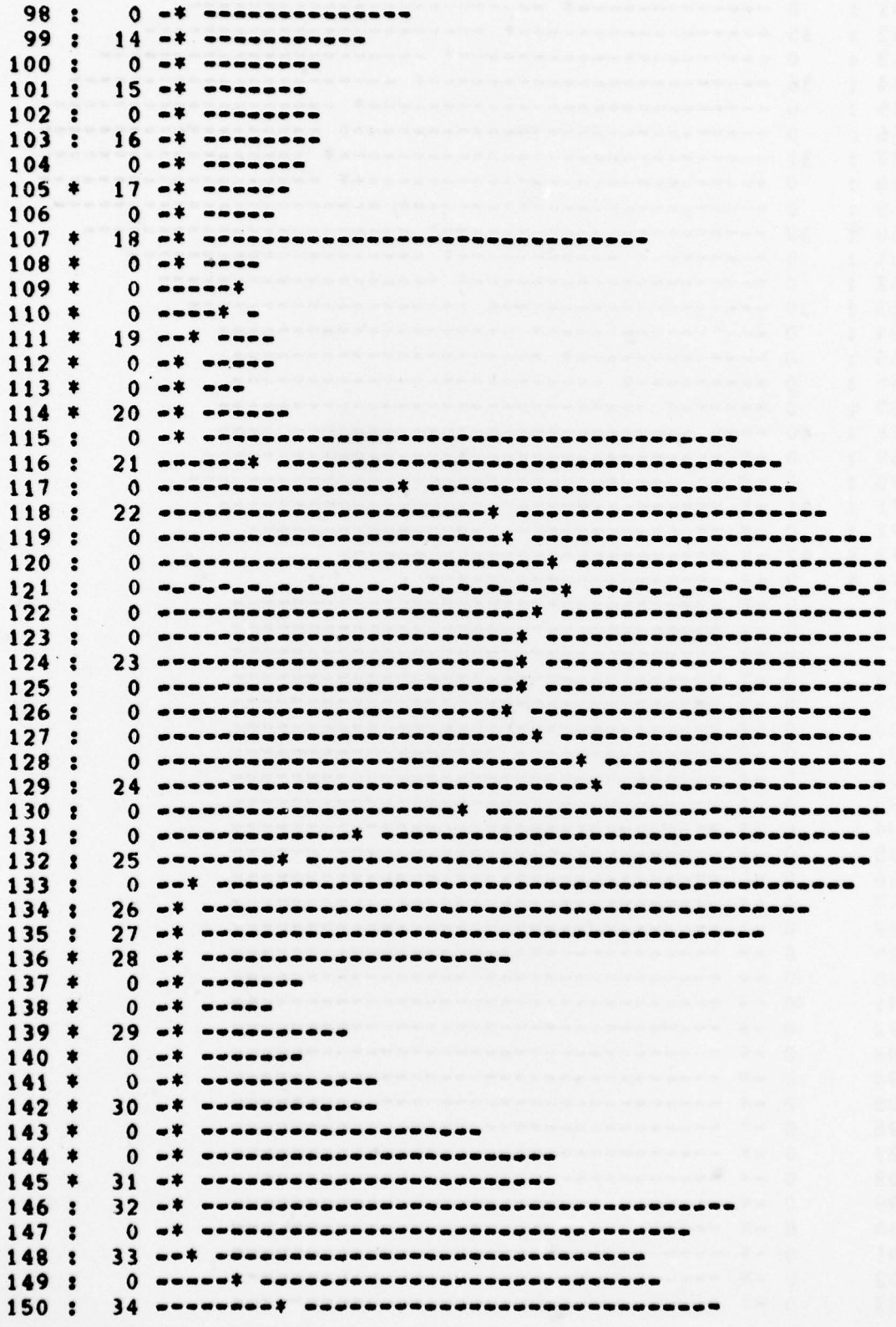


Fig. 1A. SAMPLE TRAINING PLOT, "ONE-TWO-THREE" (Second Section).

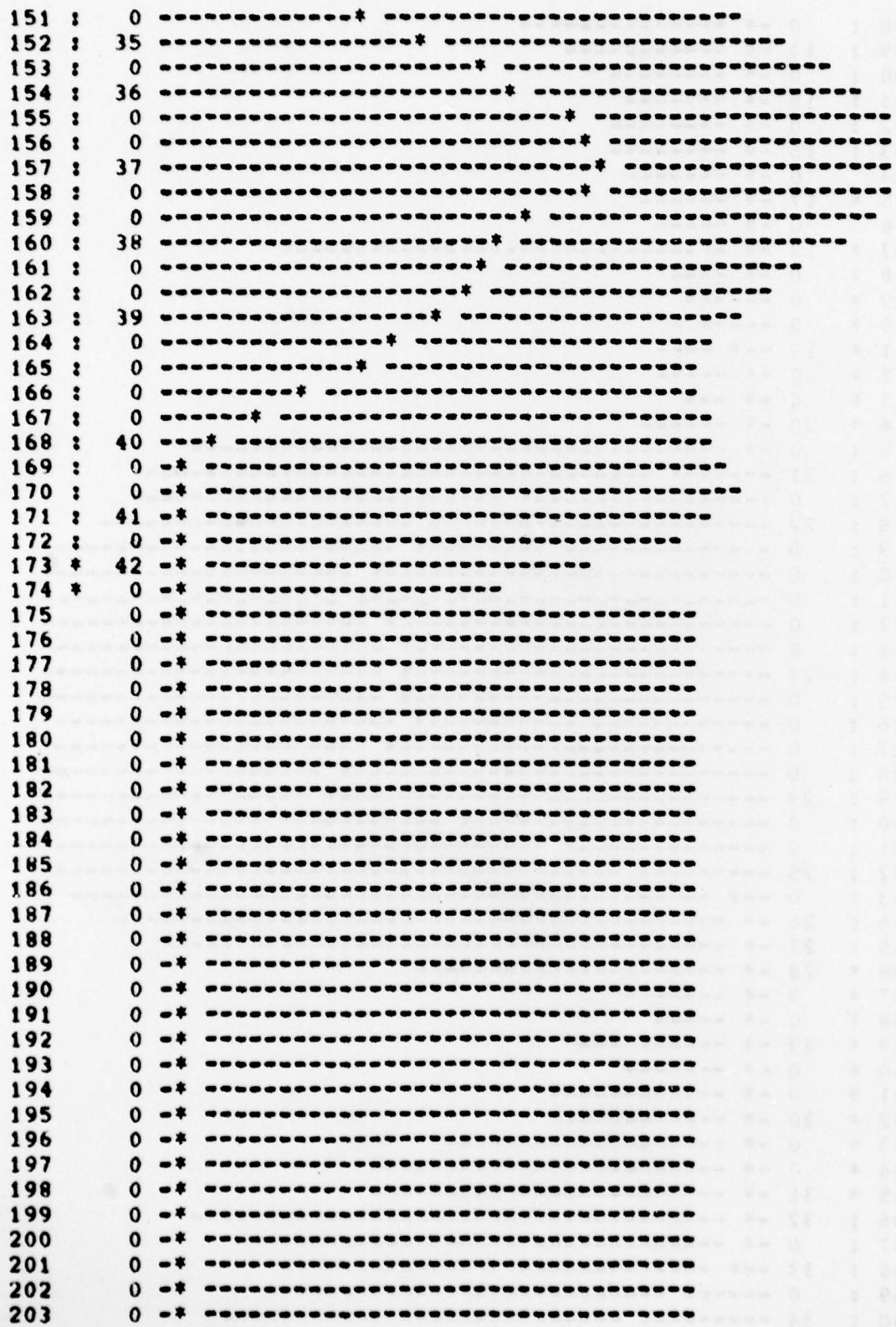


Fig. 1A. SAMPLE TRAINING PLOT, "ONE-TWO-THREE" (Third Section).

```
## PLTREC ##  RECOGNITION PLOT

CURRENT TEMPLATE FILE = CLASS02.TMP CURRENT WORD FILE = MASTER.WRD

UTTERANCE FILE = 123MHB.UTT FIRST TAPE SAMPLE =    0.00 SIZE =   0.00

FVOWEL=  12.00  FRATIO=   0.60  FACLV=0.00000500  CHANGE=  0.960 IRECTH=   130

DATE : 21-JAN-79   TIME : 20:35:41

START SAMPLE =    57   FINAL SAMPLE =   214

  57 *   1 *-------------------------------------    4AB3( 7)153 4AB2( 6)148 1AB1( 9)143 FOUR   (12)157
  58 *   0 *
  59     0 *
  60     0 *
  61     0 *
  62     0 *
  63     0 *
  64     0 *
  65     0 *
  66     0 *
  67     0 *
  68     0 *
  69     0 *
  70     0 *
  71     0 *
  72     0 *
  73     0 *
  74     0 *
  75     0 *
  76 *   2 *---------------------------------------    1AB1( 8)182 9AB2( 9)152 4AB2( 5)149 ONE    (13)157
  77     0 *
  78 :   0 *
  79 :   3 *-----------------------------------------    1AB1( 7)182 9AB2( 8)158 PAB1( 9)136 ONE    (12)167
  80 :   0 -*
  81 :   4 ---*-------------------------------------    1AB1( 6)185 9AB2( 8)157 PAB1( 8)145 ONE    ( 1)169 ONE
  82 :   5 -----*-----------------------------------    1AB1( 5)183 9AB2( 7)153 5AB1( 7)152 ONE    (10)167
  83 :   6 --------*--------------------------------    1AB1( 5)168 5AB1( 6)156 9AB2( 6)150 ONE    ( 9)159
  84 :   7 ------------*----------------------------    4BB2( 6)161 5AB1( 6)144 7BB1( 8)143 ( 0)  0
  85 :   0 -              *
  86 :   8 --------------------*--------------------    4BB2( 5)150 7BB1( 7)149 1BB1( 7)148 ( 0)  0
  87 :   0 -                  *
  88 :   9 ------------------------------*-----------    1BB1( 6)152 7BB1( 7)151 5BB1( 6)144 ( 0)  0
  89 :   0 -                    *
  90 :   0 -                   *
  91 :  10 ------------------*------------------------    7BB1( 7)152 1BB1( 5)150 5BB1( 5)147 ( 0)  0
  92 :   0 -            *
  93 :  11 --------*--------------------------------    7BB1( 6)157 1BB1( 6)141 5BB1( 5)133 ( 0)  0
  94 :   0 -     *
  95 :  12 ---*-------------------------------------    7BB1( 6)152 3AB1( 7)141 1BB1( 6)131 ( 0)  0
  96 :   0 - *
  97 :  13 --*---------------------------------------    3AB1( 6)143 7BB1( 5)138 3BB1( 4)111 ( 0)  0
  98 :   0 -*
  99 :  14 -*-------------------------------------    3AB1( 7)140 9AB1( 9)117 0AB1( 9)110 ( 0)  0
```

Fig. 1B. SAMPLE RECOGNITION PLOT, "ONE-TWO-THREE" (Section 1).

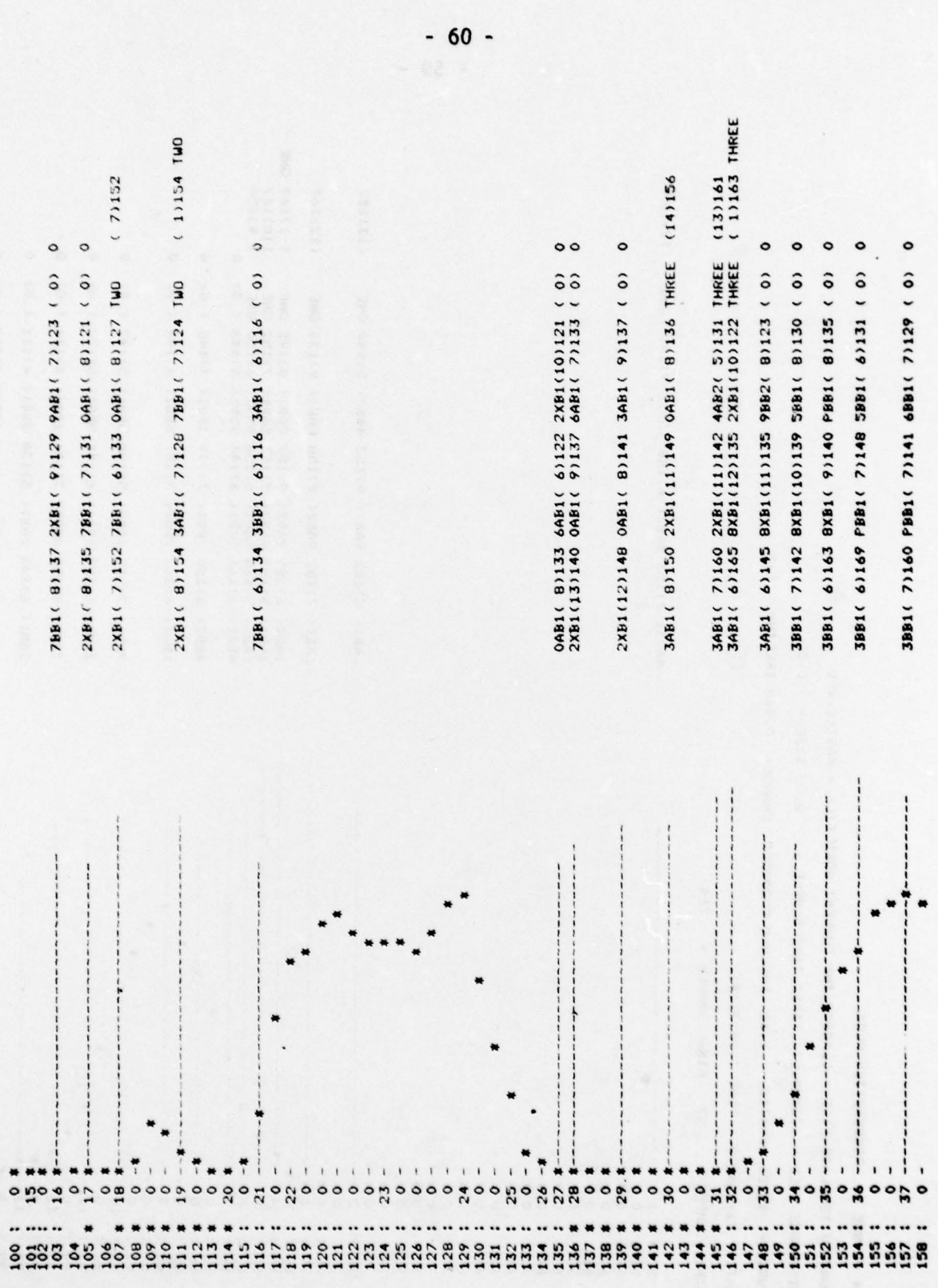


Fig. 1B. SAMPLE RECOGNITION PLOT, "ONE-TWO-THREE" (Section 2).

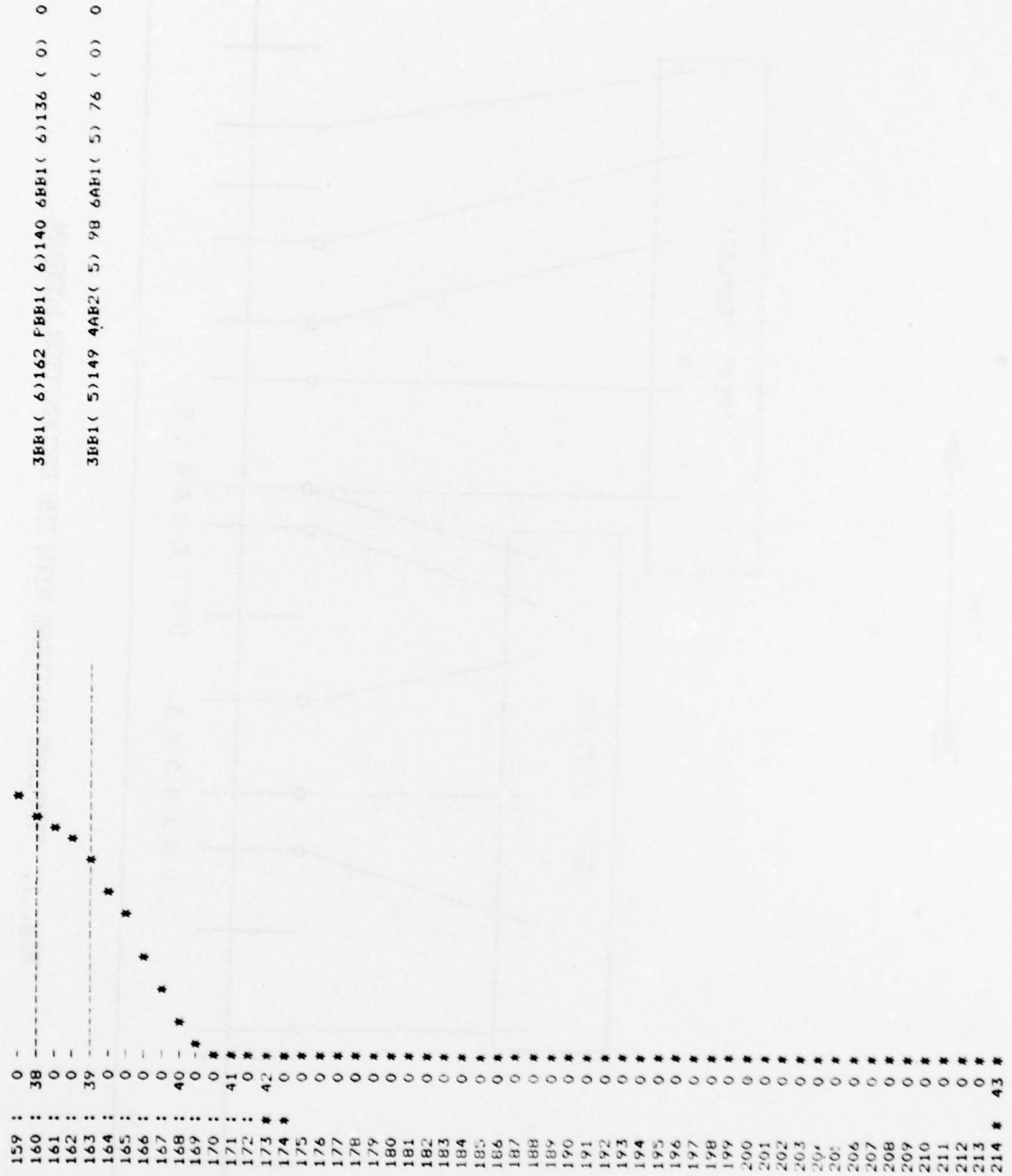


Fig. 1B. SAMPLE RECOGNITION PLOT, "ONE-TWO-THREE" (Section 3).

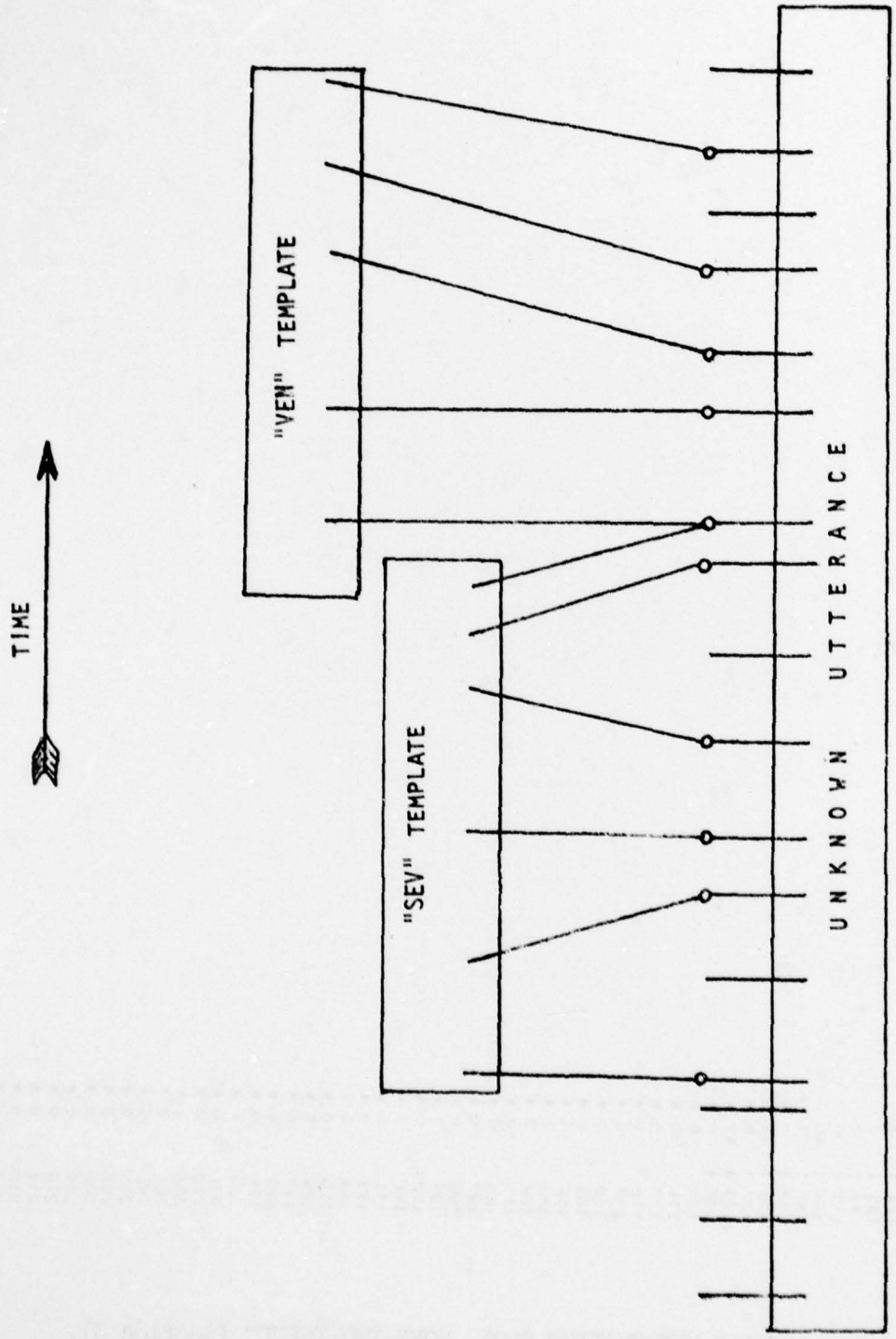


Fig. 2. TEMPLATE MATCHING WITH THE RECOGNITION PROGRAM.

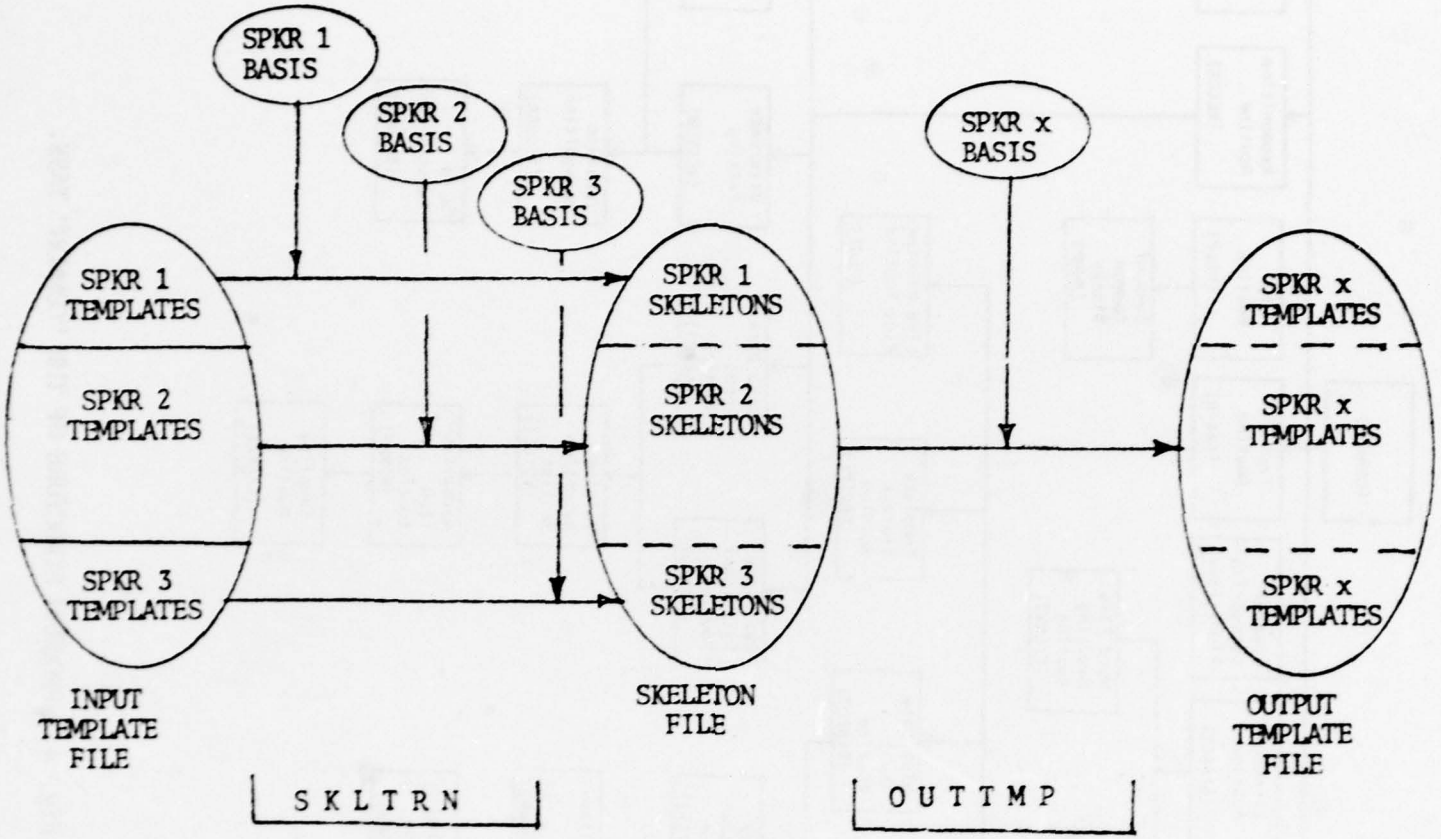


Fig. 3. SKELETON EXPANSION AND TEMPLATE RECONSTITUTION

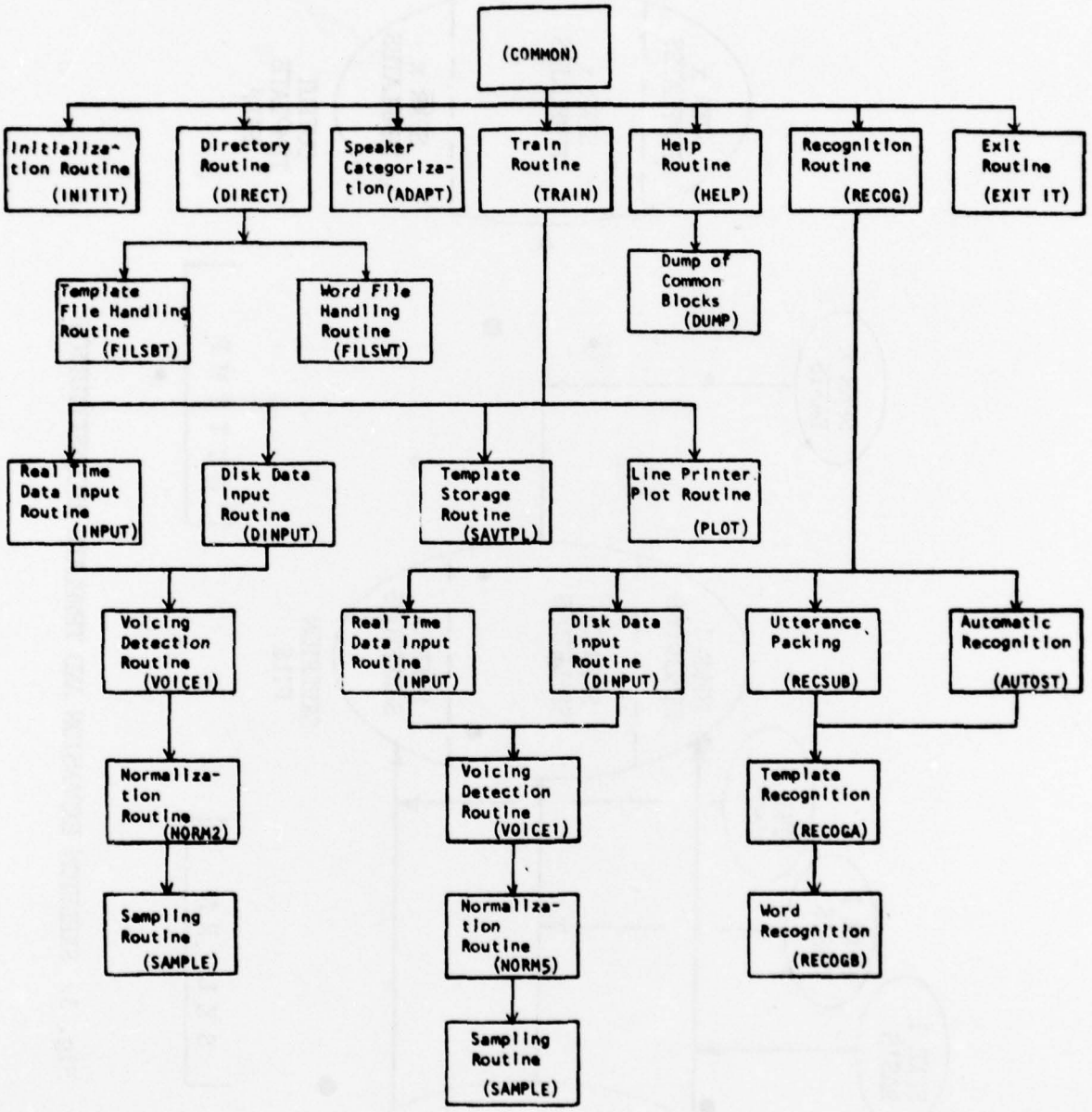


Fig. 4. PROGRAM STRUCTURE OF THE "COMMON" TASK.

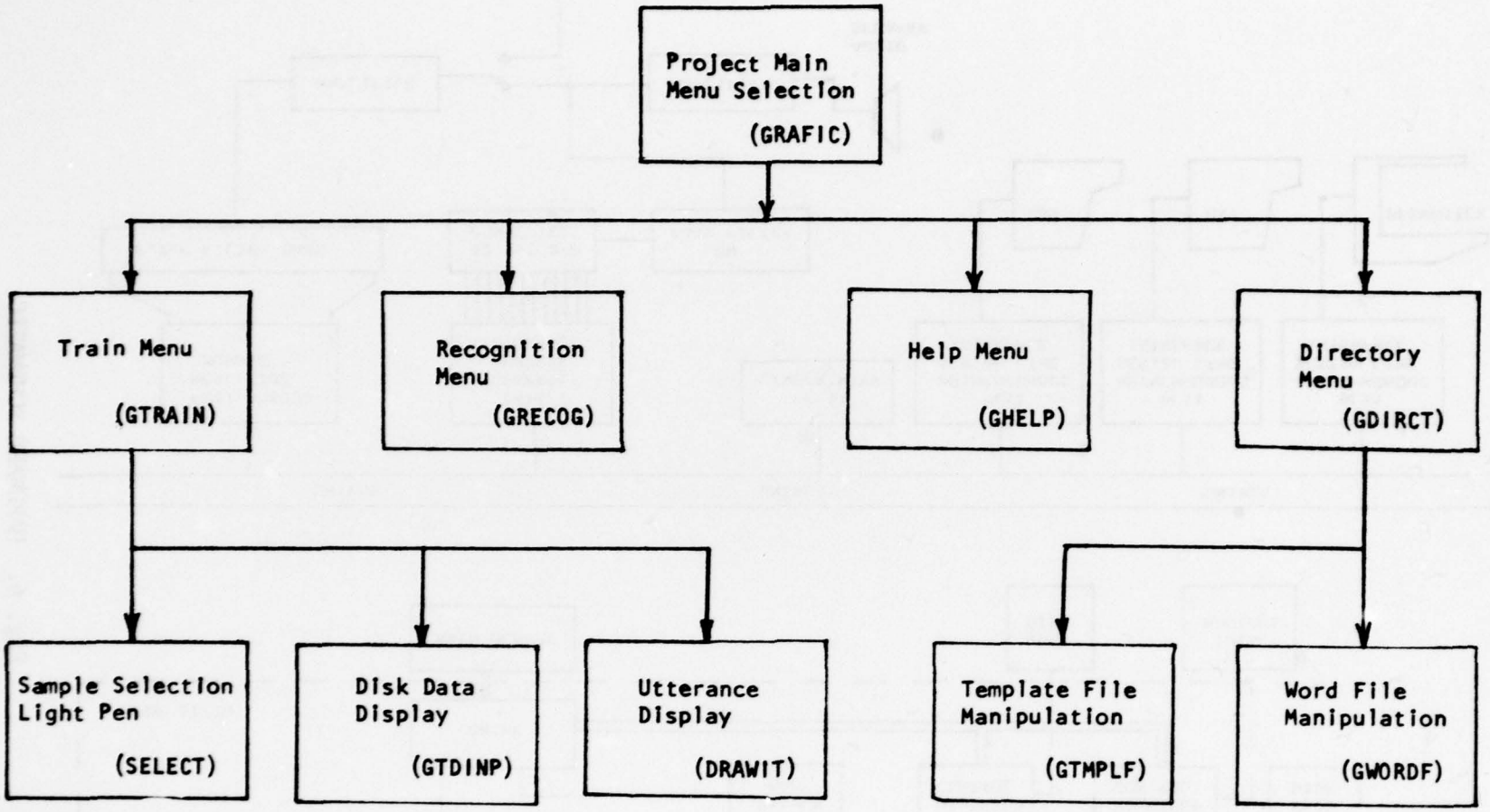


Fig. 5. "GRAFIC" TASK PROGRAM STRUCTURE

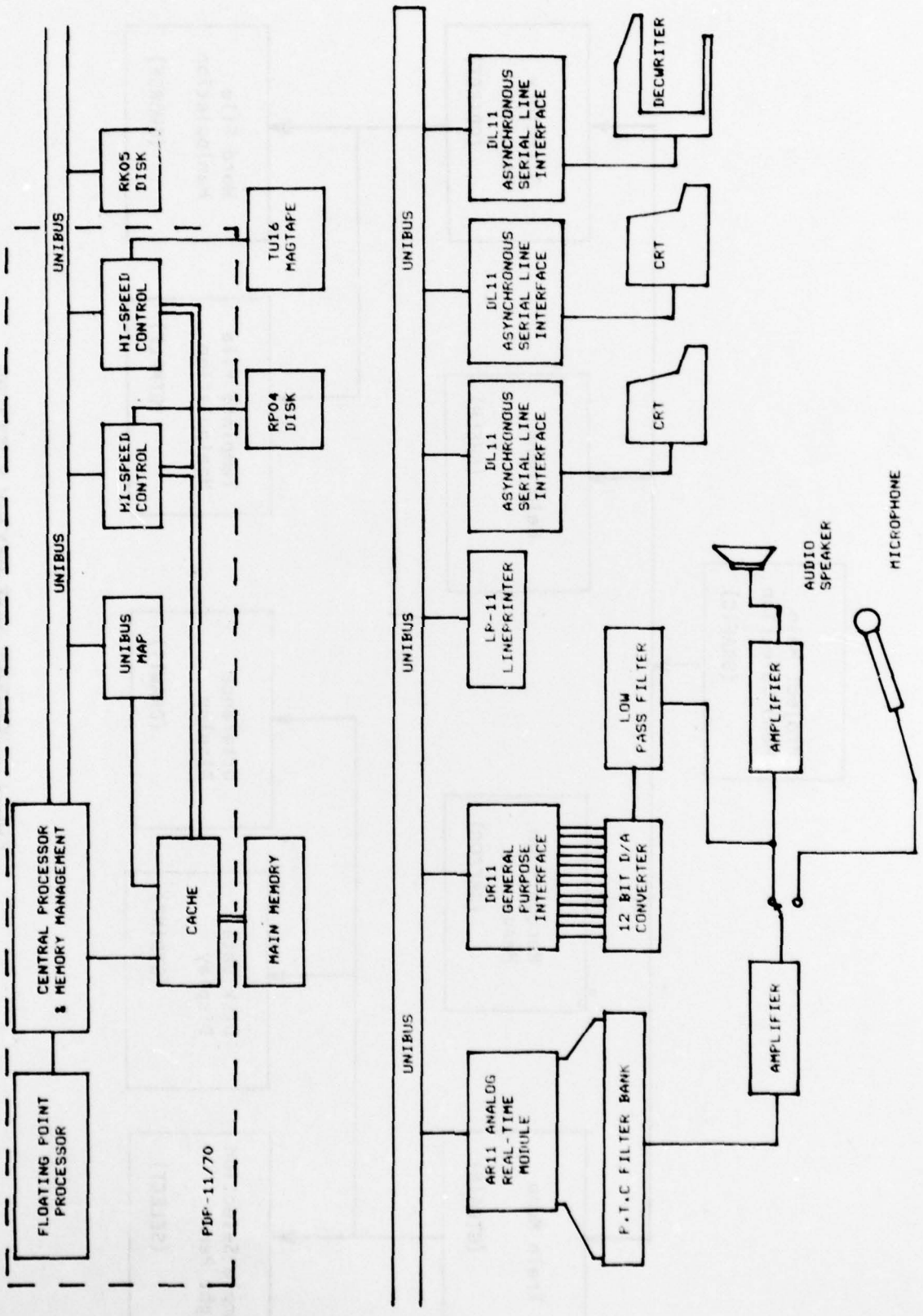


Fig. 6. HARDWARE SCHEMATIC.

MISSION
of
Rome Air Development Center

RADC plans and executes research, development, test and selected acquisition programs in support of Command, Control Communications and Intelligence (C^3I) activities. Technical and engineering support within areas of technical competence is provided to ESD Program Offices (POs) and other ESD elements. The principal technical mission areas are communications, electromagnetic guidance and control, surveillance of ground and aerospace objects, intelligence data collection and handling, information system technology, ionospheric propagation, solid state sciences, microwave physics and electronic reliability, maintainability and compatibility.